

Multi-example feature-constrained back-projection method for image super-resolution

Junlei Zhang¹, Dianguang Gai², Xin Zhang¹, and Xuemei Li¹ (✉)

© The Author(s) 2016. This article is published with open access at Springerlink.com

Abstract Example-based super-resolution algorithms, which predict unknown high-resolution image information using a relationship model learnt from known high- and low-resolution image pairs, have attracted considerable interest in the field of image processing. In this paper, we propose a multi-example feature-constrained back-projection method for image super-resolution. Firstly, we take advantage of a feature-constrained polynomial interpolation method to enlarge the low-resolution image. Next, we consider low-frequency images of different resolutions to provide an example pair. Then, we use adaptive k NN search to find similar patches in the low-resolution image for every image patch in the high-resolution low-frequency image, leading to a regression model between similar patches to be learnt. The learnt model is applied to the low-resolution high-frequency image to produce high-resolution high-frequency information. An iterative back-projection algorithm is used as the final step to determine the final high-resolution image. Experimental results demonstrate that our method improves the visual quality of the high-resolution image.

Keywords feature constraints; back-projection; super-resolution (SR)

1 Introduction

The aim of image super-resolution (SR) is to determine a corresponding high-resolution (HR)

image from one or multiple low-resolution (LR) images [1, 2]. As a basic task in image processing, image SR is widely applied in many fields, such as computer vision, medical imaging, computer animation, and digital media technology [3]. As many diverse unknown pixels are estimated from groups of pixels in the original LR image, image super-resolution still faces great challenges. Over the past several decades, many experts and scholars have undertaken much research into producing SR images. SR image algorithms can be loosely classified into three categories of methods based on interpolation, reconstruction, and example learning, respectively.

The classical method is to use polynomial interpolation, particularly using cubic splines [4, 5] or cubic convolution [6]. Interpolation-based SR methods typically utilize different kernel functions to estimate the unknown pixels in the HR image. Such methods are widely applied in a variety of commercial image processing softwares: the advantage of this class of SR approaches is that they are simple yet fast. However, the drawback is that they often blur details in textures and cause jagged artifacts along edges [7], as they do not model edges and textures in the image.

Reconstruction-based SR methods are based upon an image degradation model and solve an ill-posed inverse problem of deblurring, up-sampling, and denoising to produce the high-quality image. The iterative back-projection (IBP) method, first proposed by Irani and Peleg [8], projects errors back to the HR image iteratively, and the final HR image is the one with minimum reconstruction errors. However, the HR images produced always suffer from obvious jagged and ringing artifacts along edges, and the back-projection of reconstruction errors ignores anisotropic structures in image features.

¹ School of Computer Science and Technology, Shandong University, Jinan, 250101, China. E-mail: J. Zhang, 17865135006@163.com; X. Li, xmli@sdu.edu.cn (✉).

² Earthquake Administration of Shandong Province, China.

Manuscript received: 2016-09-09; accepted: 2016-12-22

Later, Dong et al. [9] proposed a nonlocal iterative back-projection (NLBP) method, combining nonlocal information with the IBP algorithm, which effectively reduces the reconstruction errors. Unfortunately, the NLBP algorithm may induce noise during the process of searching pixels for local information for reconstruction.

In recent years, example learning-based methods for image SR have become popular. This type of method predicts the unknown image information by learning from known instances. Example-based SR can be further generally subdivided into three categories of methods based on image pyramids, sparse representations, and neighbor embedding. Image pyramid methods obtain HR image sequences, which are taken as known examples, and according to the similarity between different resolution versions of the same image, the high-resolution image blocks are predicted [10–12]. With the gradual improvement of the theory of sparse representation [13], the method was introduced to the field of image SR, and has not been investigated by many researchers [14–17]. The principle of this kind of method is that the LR image can be sparsely expressed using a low-resolution dictionary, giving weights for use with a corresponding high-resolution dictionary to obtain the HR image. Neighbor embedding methods are based on local linear embedding [18–20], which finds several neighboring image patches for each LR image patch in the low-resolution dictionary, and calculates each neighbor's weight using a least-squares method. The weights are used to combine HR image patches to get the final HR image patches.

Compared with the polynomial interpolation method, the example-based method has higher complexity, but the resulting images are visually better at preserving image features and keeping more image details. Although these approaches are capable of adding details, the output image quality depends greatly on the image training set selected.

In this paper, we put forward a novel multi-example feature-constrained back-projection method for image super-resolution. Unlike other multiple example-based methods, image instances come from the input image instead of from an external image library. The proposed method first makes use of a simple and efficient feature interpolation algorithm to initialize the HR image. Then the instance

pair comprises the initial HR image and the low-frequency information from the low-resolution input image. For each patch in the high-resolution low-frequency image we seek similar patches from the low-resolution image using an adaptive k NN search algorithm, which learns a regression model between similar patches. This learnt model is applied to the low-resolution high-frequency image to augment it with high-resolution high-frequency information. Iterative back-projection is used as the final step to get the final HR image. Experiments indicate that the proposed method achieves highly competitive performance in visual quality, especially along the edges and within textures.

The rest of this paper is organized as follows. In Section 2, we briefly review the image degradation model, which forms the basis of this paper. We present our algorithm in Section 3. In Section 4, we introduce the feature-constrained interpolation, and the iterative back-projection algorithm is described in Section 5. Section 6 gives experimental results for our method, compares it with other state-of-the-art methods, and draws conclusions.

2 Image degradation model

The image degradation model describes the inverse process of image super-resolution, and indicates reasons for image degradation. In this paper, we use degradation model formulated as

$$L = D(B * H) + N \quad (1)$$

where L is the LR image, H is the HR image, D is downsampling by a scale factor, B is an operation which can be interpreted as Gaussian smoothing, $*$ is the convolution operation, and N is noise. Figure 1 gives a schematic diagram for downsampling by a scale factor $n = 2$, where the undegraded pixels are represented by red dots and the blurred pixels are represented by orange dots.

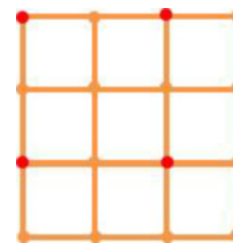


Fig. 1 Schematic diagram of degradation model.

3 Example-based method

Relative to smooth areas, the changes at edges and the texture characteristics of natural images, are obvious. The human visual system is more sensitive to high-frequency areas, so it is crucial to maintain the structure of image feature areas in SR. This paper makes full use of edges and texture structures, and other image features, with using a feature-constrained interpolation method for initialization. In addition, in images, it is often the case that many patterns appear repeatedly in the image [21–23], especially in regions with regular structures. This property is called image self-similarity, and is very helpful to fix pixels with disturbing artifacts. In this paper, we propose a multi-example feature-constrained back-projection method for SR which uses the similarity between images at different resolutions. The method is as follows.

Given an input LR image L , the aim of image super-resolution is to determine the HR image. As shown in Fig. 2, we enlarge the input image L n times using a feature-constrained polynomial interpolation method to get an initial HR image. Part of the high-frequency information is lost due to polynomial interpolation, so the initial HR image is considered to be a low-frequency image, denoted $I_{lf,hr}$ (low-frequency high-resolution image). Using a Gaussian fuzzy sampling model for image degradation of

image L , the same feature-constrained polynomial interpolation method is applied to get the same resolution image as input image L without high-frequency information, denoted $I_{lf,lr}$ (low-frequency low-resolution image). Then $\{I_{lf,hr}, I_{lf,lr}\}$ constitute the known example pair at low frequency in high- and low-resolution images. For the high-frequency information, compared to image L , $I_{lf,hr}$ is missing part of the high-frequency information, so the high frequency of the LR image is $I_{hf,lr} = L - I_{lf,lr}$. The unknown high frequency of HR image is given by $I_{hf,hr} = H - I_{lf,hr}$, with $\{I_{hf,hr}, I_{hf,lr}\}$ regarded as the example pair for high frequency at a different resolution. We then determine a regression model by learning from the example pair $\{I_{lf,hr}, I_{lf,lr}\}$, and apply it to the low-resolution high-frequency image to determine the high-resolution high-frequency information. This can be written $H_0 = I_{lf,hr} + I_{hf,hr}$. Iterative back-projection is used as the final step to get the final HR image. Figure 2 gives a schematic diagram of the proposed approach.

3.1 Feature vectors

The regression relation between the image example pair $\{I_{lf,hr}, I_{lf,lr}\}$ is based on image patches. Thus, we firstly extract image patches in order. Each small patch has 3×3 pixels. In order to increase information consistency between the image patches, the number of overlapping pixels is set to 2. The image patches make up feature vectors; the feature vector sets of the images $I_{lf,lr}$, $I_{lf,hr}$, $I_{hf,lr}$, $I_{hf,hr}$ are respectively $v_l: \{b_l^i\}_{i=1}^n$, $v_h: \{b_h^i\}_{i=1}^m$, $V_l: \{B_l^i\}_{i=1}^n$, $V_h: \{B_h^i\}_{i=1}^m$, where m is the number of HR image patches and n is the number of LR image patches.

In this paper, we employ Euclidean distance to measure similarity between feature vectors, and hence the similarity of image patches. To make feature vectors suitably reflect image features, the mean value is subtracted from the pixel values, and these are combined with weighted information from the image for use as feature vectors of the low-frequency images, $I_{lf,hr}$ and $I_{lf,lr}$. Figure 3 illustrates the feature vector representation for an image patch whose center coordinates are (46, 70). The edge information is the result of Canny edge detection, λ denotes weight, and the feature vectors of the high-frequency images $I_{hf,hr}$ and $I_{hf,lr}$ are expressed as pixel values.

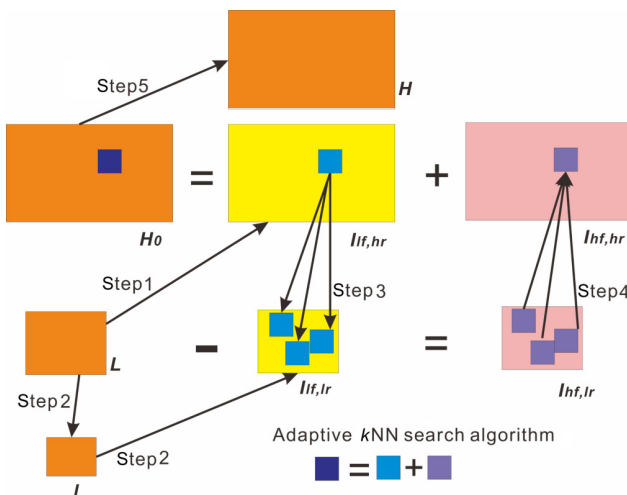


Fig. 2 Schematic diagram of the proposed approach. Steps 1 and 2 use feature-constrained polynomials, steps 3 and 4 are performed using adaptive k NN search, and step 5 uses iterative back-projection.

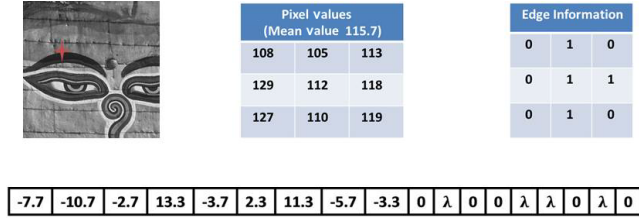


Fig. 3 Feature vectors.

3.2 Adaptive k NN search algorithm

The relationship between the example pair $\{I_{lf,hr}, I_{lf,lr}\}$ is a one-to-many adaptive multi-instance regression model. Compared to the single-instance model in Ref. [1], this model is more robust. LLE-based k NN search algorithms search for fixed k instances for each patch, but on account of the anisotropic structures of image features, we take advantage of an adaptive k NN search algorithm.

The i th image patch belonging to $I_{lf,hr}$ has feature vector b_h^i . We search adaptively for k similar vectors from the feature vector of $I_{lf,lr}$, v_l , to find a similar set for b_h^i , denoted $S^i : \{b_t\}$. Then weighted reconstruction is used to get $b_h^{i'}$.

$$b_h^{i'} = \sum_{b_t \in S^i} \omega_t b_t \quad (2)$$

We sort the feature vectors of v_l in descending order according to the degree of similarity to b_h^i , then add to $S^i : \{b_t\}$ in sequence the most similar feature vectors to b_h^i . When the reconstruction error $e = \|b_h^i - b_h^{i'}\|^2$ no longer changes, we stop adding vectors to $S^i : \{b_t\}$. The objective function used is

$$\begin{cases} \hat{t} = \arg \min \|b_h^i - b_t\|^2 \\ \text{s.t.} \quad \min \|b_h^i - b_h^{i'}\|^2 \end{cases} \quad (3)$$

In the LEE-based k NN search method [18], the weight values for similar patches are calculated by a least-squares method, and their values sum to 1, which results in appearance of negative weights. Thus the image quality usually fluctuates with the value of k . To avoid this problem, we use a Gaussian function to set the weight values of similar patches, making them all positive:

$$\begin{cases} w_t = \exp(-(\|b_h^i - b_t\|_2)/Nh^2) \\ \omega_t = w_t / \sum w_t \end{cases} \quad (4)$$

where N is the size of each image patch, and h^2 controls the decay speed, set to 1 in this paper. The procedure for adaptive k NN search algorithm is summarized in Algorithm 1.

The adaptive k NN search method determines S^i ,

Algorithm 1 Adaptive k NN search algorithm

Input: b_h^i belonging to $I_{lf,hr}$; the feature vector set $v_l : \{b_t\}_{t=1}^n$.

Output: The similar set $S^i : \{b_t\}_{t=1}^k$ with b_h^i and corresponding weight values $\{\omega_t\}_{t=1}^k$.

- 1: Initialize: $k = 1$, $S^i = \emptyset$;
- 2: Calculate the degree of similarity of b_h^i and each vector in v_l , sort the feature vectors for v_l in descending order, make v_l become v_l' ;
- 3: Calculate $b_h^{i'}$ from Eq. (2) where ω_t and w_t are found from Eq. (4);
- 4: Calculate the error between $b_h^{i'}$ and b_h^i : $e_1 = \|b_h^i - b_h^{i'}\|^2$;
- 5: $d_{\text{error}} \leftarrow e_1$;
- 6: **while** $d_{\text{error}} > 0$ **do**
- 7: $k \leftarrow k + 1$;
- 8: Repeat step 3 with the new k , calculate the new error $e_2 = \|b_h^i - b_h^{i'}\|^2$;
- 9: $d_{\text{error}} \leftarrow e_1 - e_2$;
- 10: $e_1 \leftarrow e_2$;
- 11: **end while**
- 12: $k \leftarrow k - 1$.

the similar set for b_h^i , and the values of the weights. The learnt model is applied to the example pair $\{I_{hf,hr}, I_{hf,lr}\}$, so the high-frequency information in $I_{hf,hr}$, defined as B_h^i corresponding with b_h^i , can be found as follows:

$$B_h^i = \sum_{t=1}^k \omega_t B_t \quad (5)$$

By traversing the feature vector set, we get the vector set for $I_{hf,hr}$, defined as $V_h : \{B_h^i\}_{i=1}^m$. By performing the process of fetching image patches and the inverse process of vector operations, we get the $I_{hf,hr}$. As a result, the HR image can be defined as $H_0 = I_{hf,hr} + I_{lf,hr}$.

4 Feature-constrained polynomial interpolation method

In the degradation model in Eq. (1), the high-frequency information is filtered out by a Gaussian blurring filter, producing L from H . Thus, different HR images having different high-frequency information but the same low-frequency information can produce the same LR image. Thus, the initial HR image is essential to the final HR image. Most learning-based methods adopt polynomial interpolation to initialize the HR image

for simplicity, but these methods often cause severe jagged artifacts along edges and blurring as they do not consider feature areas sufficiently. Instead, this paper adopts a simple and efficient feature-constrained polynomial interpolation (FCI) method.

For every small region, a 3×3 patch shown in Fig. 4, we assume that the surface can be expressed by a quadratic function $f_{i,j}$, where (i, j) is the coordinate of the center pixel $P_{i,j}$ of the surface:

$$f_{i,j}(x, y) = a_1x^2 + a_2xy + a_3y^2 + a_4x + a_5y + P_{i,j} \quad (6)$$

where (x, y) is the coordinate for the sample point on the surface with (i, j) as the original point, $(x, y) \in [-1.5, 1.5]$, (a_1, \dots, a_5) are the unknown coefficients, and the coordinates of $P_{i,j}$ are $(0, 0)$.

As shown in Fig. 4, there are four directions marked by red arrows around the center pixel $P_{i,j}$ and other eight pixels surrounding it. The first order differences at the original point along the four directions x , y , $x + y$, and $x - y$ are correspondingly as follows:

$$\begin{cases} x : a_4 = (P_{i+1,j} - P_{i-1,j}) / 2 \\ y : a_5 = (P_{i,j+1} - P_{i,j-1}) / 2 \\ x + y : a_4 + a_5 = (P_{i+1,j+1} - P_{i-1,j-1}) / 2 \\ x - y : a_4 - a_5 = (P_{i+1,j-1} - P_{i-1,j+1}) / 2 \end{cases} \quad (7)$$

Coefficients a_4 and a_5 can be computed by solving Eq. (7) with a weighted least-squares method defined as follows:

$$(a_4, a_5) = \arg \min \sum_1^4 W_i U_{q_i}^2 \quad (8)$$

where U_{q_i} can be expressed via the following set of equations:

$$\begin{cases} U_x = a_4 - (P_{i+1,j} - P_{i-1,j}) / 2 \\ U_y = a_5 - (P_{i,j+1} - P_{i,j-1}) / 2 \\ U_{x+y} = a_4 + a_5 - (P_{i+1,j+1} - P_{i-1,j-1}) / 2 \\ U_{x-y} = a_4 - a_5 - (P_{i+1,j-1} - P_{i-1,j+1}) / 2 \end{cases} \quad (9)$$

The weight W_i for each equation is given by

$$W_i = \frac{\sigma + \Delta_{i\perp}^2}{\sigma + \Delta_i^2} \quad (10)$$

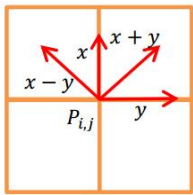


Fig. 4 Four directions.

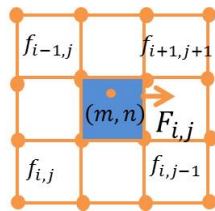


Fig. 5 Overlap region for four neighboring patches.

where Δ_i is the second order difference along the same direction as the relevant equation, and $\Delta_{i\perp}$ is the second order difference in the vertical direction. For example, Δ_i in the x direction is given by $\Delta_x = P_{i+1,j} + P_{i-1,j} - 2P_{i,j}$, and its $\Delta_{i\perp}$ is given by $\Delta_{x\perp} = P_{i,j+1} + P_{i,j-1} - 2P_{i,j}$. Here, σ is a small value to make W_i meaningful when $\Delta_i = 0$. The smaller Δ_i is, the more the possible pixels forming the edges. The relationship for $\Delta_{i\perp}$ is just the opposite.

To determine the unknown coefficients a_1, a_2, a_3 , we fit the other eight pixels around $P_{i,j}$ using the objective function:

$$f_{i,j}(u, v) = a_1u^2 + a_2uv + a_3v^2 + a_4u + a_5v + P_{i,j} \quad (11)$$

where (u, v) is the coordinate of the pixel around $P_{i,j}$, $(u, v) \in [-1, 0, 1]$, and $u^2 + v^2 \neq 0$. Abbreviating the function in Eq. (7), we can write that equation as $f_{i,j}(u, v) = P_{i+u,j+v}$. The equation is solved by using the least-squares method on the following function:

$$\arg \min \sum_{u,v \in [-1,0,1], u^2+v^2 \neq 0} W_{uv} (f_{i,j}(u, v) - P_{i+u,j+v})^2 \quad (12)$$

where weights along the same direction are defined in the same way as for Eq. (6). For example, pixels $P_{i+1,j}$ and $P_{i-1,j}$ both lie in the same direction along the x direction, so the weight can be set to

$$W_{10} = W_{-10} = W_x \quad (13)$$

We can observe from Fig. 5 that the blue region, which is the center of the overlap region for four neighboring patches, can be defined as

$$F_{i,j} = (1 - m)(1 - n)f_{i,j} + m(1 - n)f_{i+1,j} + (1 - m)nf_{i,j+1} + mnf_{i+1,j+1} \quad (14)$$

where (m, n) is the coordinate of the interpolated point for which (i, j) is the original point. Using function F , we can interpolate unknown pixels by averaging four adjacent pixels with relevant scale factors.

5 Iterative back-projection

After these operations are complete, we have the HR image H_0 . In order to further improve the quality of the image, we introduce iterative back-projection as a global post-processing operation. In fact, it plays an important role in determining the HR image's visual quality.

In ideal conditions, a perfect HR image will reproduce the same LR image as the original one

by the degradation model in Eq. (1). However, the HR image always reproduces LR images with errors compared to the input LR image. To describe this concisely, let L denote the input LR image and H denote the reconstructed HR image. The ideal HR image is the one with minimum reconstruction error:

$$\begin{cases} \hat{H} = \arg \min \|L - D(B * H)\|_2^2 \\ \text{s.t.} \quad D(B * H) = L \end{cases} \quad (15)$$

This problem is solved by projecting reconstruction errors back to the HR image iteratively:

$$H^{t+1} = H^t + P(L - D(B * H^t)) \quad (16)$$

where $P(\cdot)$ is feature-constrained polynomial interpolation as explained in Section 3; we use it to avoid propagating errors isotropically during the iterative process which would result in jagged artifacts and ringing effects in the HR image. The back-projection procedure is summarized in Algorithm 2.

Here, the number of iterations is set to 2. This algorithm provides an improved HR image. Then, we shrink the improved HR image using Eq. (1), giving an LR image denoted L' . For each pixel p'_i in L' and the corresponding pixel p_i in the input LR image L , we assume there is a related scale coefficient $\theta_i = p_i / p'_i$. Using the inverse process of Eq. (1), we can find 4 pixels in the improved HR image by iterative back-projection for each corresponding pixel in L' . Then we get values for these 4 pixels by multiplying them by the relevant scale coefficient θ_i , finally giving the desired HR image.

6 Experimental results and conclusions

In this section we assess the effectiveness of the proposed method (feature-constrained multi-

example back-projection, FCMEBP) through experiments, and compare it with other five methods. CSFI [24] is cubic surface fitting with edges as constraints. NeedFS [25] is based on neighbor embedding edge detection feature selection. IUIE [26] and NARM [27] represent example-based and sparse coding methods respectively. FCME is our multi-example feature-constrained method ignoring back-projection.

We use the 8 common test images in Fig. 6. The test images are regarded as reference HR images, and appropriate LR images are determined from these HR images using a scaling factor of $n = 2$. The six super-resolution methods are used to magnify the LR images to the same resolution as the original HR images. The effectiveness of each method is verified by comparing the results with test HR images quantitatively and visually. To be fair, for each method, the degradation model is set in accordance with the corresponding reference.

6.1 Quantitative assessment

In order to evaluate the quality of the results of magnification, we adopt the most commonly used objective methods based on comparisons with explicit numerical criteria [28], including peak signal to noise ratio (PSNR) and structural similarity (SSIM). PSNR measures the difference between the HR image and the test image, while SSIM measures the similarity of the structural information in the images. The numerical value of PSNR for each standard image is listed in Table 1, and the value of SSIM of each test image is listed in Table 2. Table 1 shows that the proposed method achieves higher PSNR than the other five methods. For SSIM, the proposed method, FCME, and NeedFS have about

Algorithm 2 Iterative back-projection

Input: HR image H_0 .

Output: The final HR image H .

- 1: Initialize counter: $t = 0$;
 - 2: $H^t = H_0$;
 - 3: **while** $t < 2$ **do**
 - 4: Find reconstruction error: $e = L - D(B * H^t)$;
 - 5: Upscale reconstruction error: $e' = P(e)$;
 - 6: Update H : $H^{t+1} = H^t + e'$;
 - 7: $t = t + 1$;
 - 8: **end while**
-

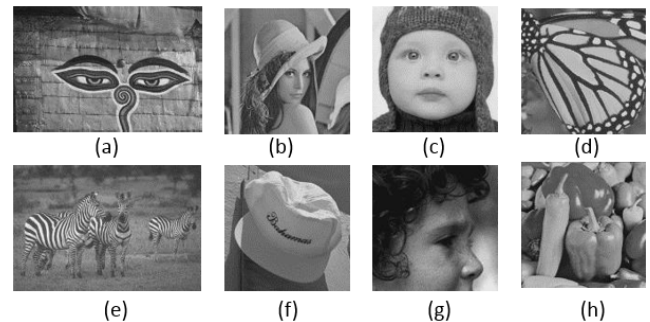


Fig. 6 Test images. Top: Artwall, Lena, Baby, Butterfly. Bottom: Zebra, Hat, Head, Pepper.

Table 1 PSNR values

Image	CSFI	NeedFS	IUIE	NARM	FCME	Ours
Artwall	26.88	26.38	25.73	26.62	27.43	27.57
Lena	34.98	34.26	33.53	35.31	35.54	35.68
Butterfly	28.27	29.63	26.11	30.27	30.17	30.39
Baby	37.19	36.03	34.61	36.72	37.37	37.55
Zebra	25.38	25.30	24.04	25.63	26.41	26.57
Hat	32.18	32.07	30.88	33.16	33.52	33.58
Pepper	35.25	32.23	33.88	34.80	35.82	36.01
Head	34.88	33.85	33.99	34.37	34.99	35.07
Average	31.89	31.22	30.35	32.11	32.66	32.80

Table 2 SSIM values

Image	CSFI	NeedFS	IUIE	NARM	FCME	Ours
Artwall	0.81	0.82	0.78	0.81	0.83	0.83
Lena	0.99	0.99	0.96	0.92	0.99	0.99
Butterfly	0.93	0.95	0.91	0.95	0.95	0.96
Baby	0.99	0.99	0.98	0.95	0.99	0.99
Zebra	0.84	0.86	0.82	0.85	0.85	0.86
Hat	0.90	0.92	0.90	0.92	0.92	0.92
Pepper	0.99	0.99	0.98	0.91	0.99	0.99
Head	0.86	0.87	0.85	0.86	0.87	0.87
Average	0.91	0.92	0.90	0.90	0.92	0.92

the same values, but there is an obvious improvement compared with the other three methods.

6.2 Visual quality comparison

To compare the visual quality of each method, we illustrate two-time reconstructed HR images for the Artwall, Lena, and Pepper images in Figs. 7–9. For reasons of space, we simply show some local effects on each image, produced by each method. We see that CSFI avoids jagged artifacts effectively, but the loss of some high frequencies means that the result images are blurred, e.g., the regions surrounded by

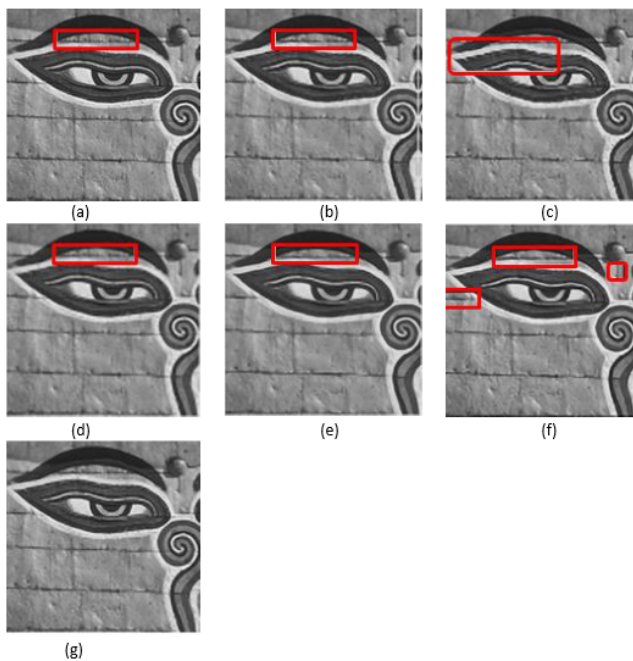


Fig. 7 Two-time reconstructed image Artwall using the six SR methods. (a) Input test image. (b) FCME. (c) NARM. (d) CSFI. (e) NeedFS. (f) IUIE. (g) Our method.

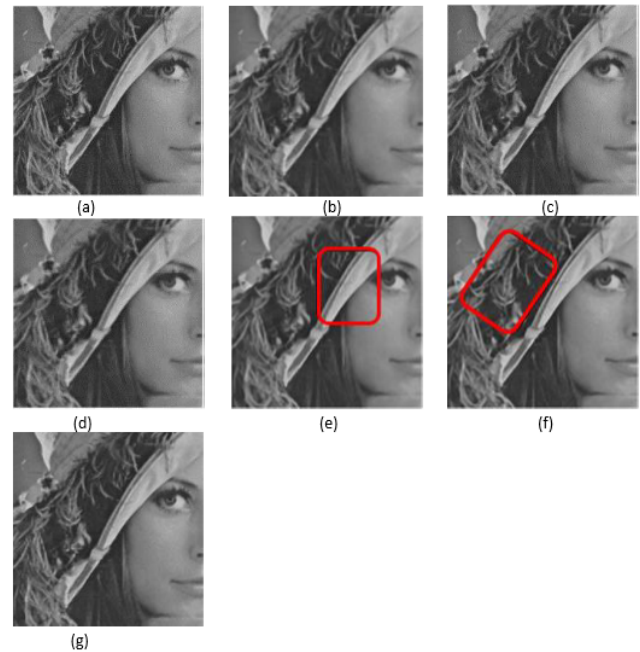


Fig. 8 Two-time reconstructed image Lena using the six SR methods. (a) Input test image. (b) FCME. (c) NARM. (d) CSFI. (e) NeedFS. (f) IUIE. (g) Our method.

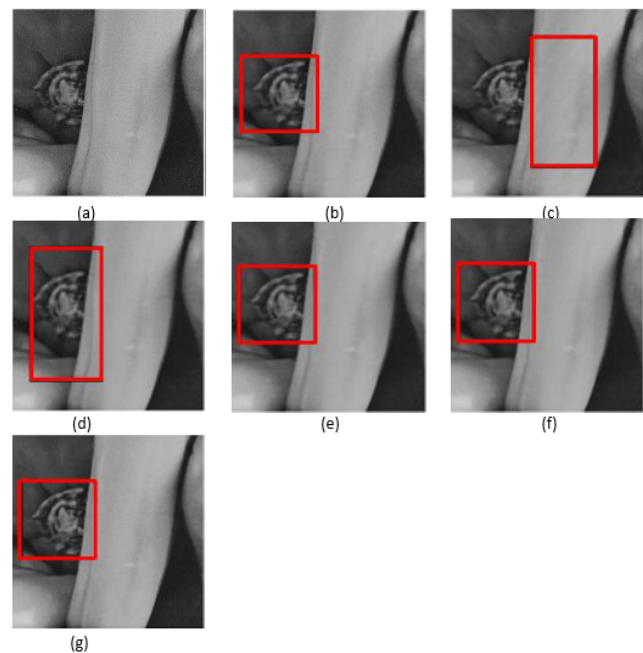


Fig. 9 Two-time reconstructed image Pepper using the six SR methods. (a) Input test image. (b) FCME. (c) NARM. (d) CSFI. (e) NeedFS. (f) IUIE. (g) Our method.

red rectangles in Fig. 7(d) and Fig. 9(d). NeedFS produces evidently blurred textures, e.g., see the Artwall's crack in Fig. 7(e) and the Pepper's handle in Fig. 9(e). IUIE produces relatively clear HR images, but edge information is distorted, e.g., in the Artwall's texture in Fig. 7(f), Lena's hair in Fig. 8(f),

and the region marked by a red rectangle in Fig. 9(f). NARM produces HR images with clear textures, but we can clearly see the blurred eyes in the Artwall image in Fig. 7(c); in Fig. 9(c), the surface texture of the Pepper is distorted. FCME produces high-frequency information which makes the HR images relatively sharp, but in Fig. 7(b) and Fig. 9(b), we note that the method cannot produce well-ordered textures.

We also compare results produced from MR brain images in Fig. 10. They show that the proposed method works well not only for natural images, but also for MR images, producing sharp edges while effectively avoiding jagged artifacts during the SR process.

In order to further compare the FCME and FCMEBP methods, we illustrate other two-time reconstructed HR images for the Baby and Butterfly in Figs. 11 and 12, respectively. In Fig. 11(b) and Fig. 12(b), information loss along the edge of the woolly hat and the scales on the wings cause the HR images to lack details. However, the proposed method effectively regenerates image features, suppresses jagged and ringing effects, producing high-resolution information that makes the HR image clearer.

6.3 Conclusions

This paper presents a novel method for image super-

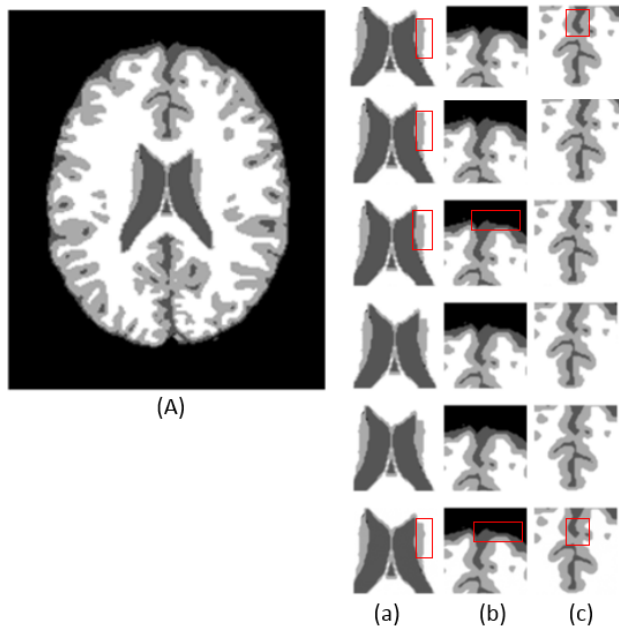


Fig. 10 Two-time reconstructed MR brain image using the six SR methods NARM, CSFI, NeedFS, IUIE, FCME, and our method, in order.

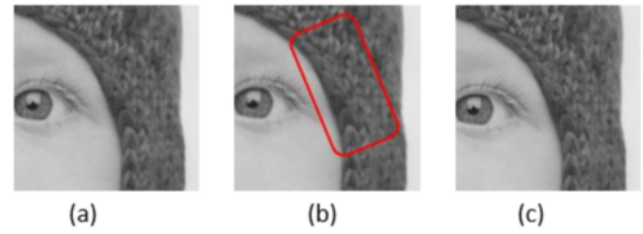


Fig. 11 Two-time reconstructed image Baby using two SR methods. (a) Input test image. (b) FCME. (c) Our method.

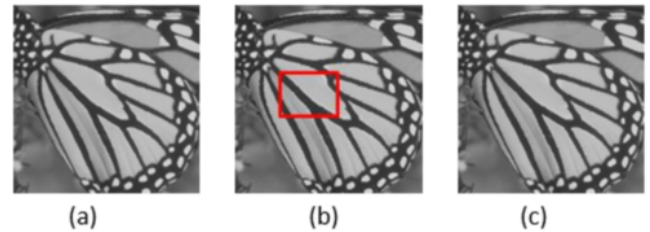


Fig. 12 Two-time reconstructed image Butterfly using two SR methods. (a) Input test image. (b) FCME. (c) Our method.

resolution based on multiple examples, using feature-constrained interpolation and back-projection. Our proposed method first obtains an HR image by using feature-constrained polynomial interpolation. We consider low-frequency images of different resolution images as the example pair. We use adaptive k NN search to find similar patches from the low-resolution image for every image patch in the high-resolution low-frequency image, allowing us to learn a regression model between similar patches. This model is applied to the low-resolution high-frequency image to get high-resolution high-frequency information. Iterative back-projection is used as the final step to get the final high-resolution image. Our experimental results demonstrate that our proposed method can achieve high-quality image super-resolution. Use of direct interpolation helps to avoid jagged artifacts and iterative back-projection preserves sharp edges.

Acknowledgements

The authors would like to thank the anonymous reviewers for giving valuable suggestions that greatly improved the paper. The authors also thank other researchers who provided the code for their algorithms for comparative testing. This project was supported by the National Natural Science Foundation of China (Grant Nos. 61572292, 61332015, 61373078, and 61272430), and the National Research Foundation for the Doctoral

Program of Higher Education of China (Grant No. 20110131130004).

References

- [1] Glasner, D.; Bagon, S.; Irani, M. Super-resolution from a single image. In: Proceedings of the IEEE 12th International Conference on Computer Vision, 349–356, 2009.
- [2] Park, S. C.; Park, M. K.; Kang, M. G. Super-resolution image reconstruction: A technical overview. *IEEE Signal Processing Magazine* Vol. 20, No. 3, 21–36, 2003.
- [3] Kolte, R.; Arora, A. Image super-resolution. Available at <https://pdfs.semanticscholar.org/20de/2880a4196a733314252a717f1a55f5f0ea64.pdf>.
- [4] Hou, H.; Andrews, H. Cubic splines for image interpolation and digital filtering. *IEEE Transactions on Acoustics, Speech, and Signal Processing* Vol. 26, No. 6, 508–517, 1978.
- [5] McKinley, S.; Levine, M. Cubic spline interpolation. *College of the Redwoods* Vol. 45, No. 1, 1049–1060, 1998.
- [6] Keys, R. Cubic convolution interpolation for digital image processing. *IEEE Transactions on Acoustics, Speech, and Signal Processing* Vol. 29, No. 6, 1153–1160, 1981.
- [7] Wang, H.; Gao, X.; Zhang, K.; Li, J. Single-image super-resolution using active-sampling Gaussian process regression. *IEEE Transactions on Image Processing* Vol. 25, No. 2, 935–948, 2016.
- [8] Irani, M.; Peleg, S. Improving resolution by image registration. *CVGIP: Graphical Models and Image Processing* Vol. 53, No. 3, 231–239, 1991.
- [9] Dong, W.; Zhang, L.; Shi, G.; Wu, X. Nonlocal back-projection for adaptive image enlargement. In: Proceedings of the 16th IEEE International Conference on Image Processing, 349–352, 2009.
- [10] Adelson, E. H.; Anderson, C. H.; Bergen, J. R.; Burt, P. J.; Ogden, J. M. Pyramid methods in image processing. *RCA Engineer* Vol. 29, No. 6, 33–41, 1984.
- [11] Bevilacqua, M.; Roumy, A.; Guillemot, C.; Alberi-Morel, M. L. Low-complexity single-image super-resolution based on nonnegative neighbor embedding. In: Proceedings of British Machine Vision Conference, 135.1–135.10, 2012.
- [12] Yang, C.-Y.; Huang, J.-B.; Yang, M.-H. Exploiting self-similarities for single frame super-resolution. In: *Computer Vision-ACCV 2010*. Kimmel, R.; Klette, R.; Sugimoto, A. Eds. Springer Berlin Heidelberg, 497–510, 2010.
- [13] Yang, J.; Wright, J.; Huang, T. S.; Ma, Y. Image super-resolution via sparse representation. *IEEE Transactions on Image Processing* Vol. 19, No. 11, 2861–2873, 2010.
- [14] Dong, W.; Shi, G.; Zhang, L.; Wu, X. Super-resolution with nonlocal regularized sparse representation. In: Proceedings of SPIE7744, Visual Communications and Image Processing, 77440H, 2010.
- [15] Yang, J.; Wright, J.; Huang, T.; Ma, Y. Image super-resolution as sparse representation of raw image patches. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1–8, 2008.
- [16] Zhang, H.; Zhang, Y.; Huang, T. S. Efficient sparse representation based image super resolution via dual dictionary learning. In: Proceedings of the IEEE International Conference on Multimedia and Expo, 1–6, 2011.
- [17] Zhao, Y.; Yang, J.; Zhang, Q.; Song, L.; Cheng, Y.; Pan, Q. Hyperspectral imagery super-resolution by sparse representation and spectral regularization. *EURASIP Journal on Advances in Signal Processing* Vol. 2011, 87, 2011.
- [18] Chang, H.; Yeung, D.-Y.; Xiong, Y. Super-resolution through neighbor embedding. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, I, 2004.
- [19] Gao, X.; Zhang, K.; Tao, D.; Li, X. Image super-resolution with sparse neighbor embedding. *IEEE Transactions on Image Processing* Vol. 21, No. 7, 3194–3205, 2012.
- [20] Roweis, S. T.; Saul, L. K. Nonlinear dimensionality reduction by locally linear embedding. *Science* Vol. 290, No. 5500, 2323–2326, 2000.
- [21] BenAbdelkader, C.; Cutler, R.; Nanda, H.; Davis, L. EigenGait: Motion-based recognition of people using image self-similarity. In: *Audio- and Video-Based Biometric Person Authentication*. Bigun, J.; Smeraldi, F. Eds. Springer Berlin Heidelberg, 284–294, 2001.
- [22] Church, K. W.; Helfman, J. I. Dotplot: A program for exploring self-similarity in millions of lines of text and code. *Journal of Computational and Graphical Statistics* Vol. 2, No. 2, 153–174, 1993.
- [23] Shechtman, E.; Irani, M. Matching local self-similarities across images and videos. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 1–8, 2007.
- [24] Caiming, Z.; Xin, Z.; Xuemei, L.; Fuhua, C. Cubic surface fitting to image with edges as constraints. In: Proceedings of the IEEE International Conference on Image Processing, 1046–1050, 2013.
- [25] Chan, T.-M.; Zhang, J.; Pu, J.; Huang, H. Neighbor embedding based super-resolution algorithm through edge detection and feature selection. *Pattern Recognition Letters* Vol. 30, No. 5, 494–502, 2009.
- [26] Freedman, G.; Fattal, R. Image and video upscaling from local self-examples. *ACM Transactions on Graphics* Vol. 30, No. 2, Article No. 12, 2011.
- [27] Dong, W.; Zhang, L.; Lukac, R.; Shi, G. Sparse representation based image interpolation with nonlocal autoregressive modeling. *IEEE Transactions on Image Processing* Vol. 22, No. 4, 1382–1394, 2013.

- [28] Hore, A.; Ziou, D. Image quality metrics: PSNR vs. SSIM. In: Proceedings of the 20th International Conference on Pattern Recognition, 2366–2369, 2010.



Junlei Zhang received his B.S. degree in computer science and technology from Shandong Jiaotong University, Jinan, China, in 2015. Currently, he is a master candidate in the School of Computer Science and Technology, Shandong University, Jinan, China. His research interests include computer

graphics and image processing.



Dianguang Gai received his master of engineering degree in computer science and technology from Shandong University, Jinan, China, and is working in the Earthquake Administration of Shandong Province. His research interests include data warehousing and earthquake prediction.



Xin Zhang is a Ph.D. student in the Department of Computer Science and Technology, Shandong University, Jinan, China. She received her bachelor degree in computer science from Shandong University in 2012. Her research interests include image processing, computer graphics,

geometry processing, and CAGD.



Xuemei Li received her master and doctor degrees from Shandong University, Jinan, China, in 2004 and 2010, respectively. She is currently an associate professor in the School of Computer Science and Technology, Shandong University, and a member of the GD and IV Lab. She is engaged

in research on geometric modeling, CAGD, medical image processing, and information visualization.

Open Access The articles published in this journal are distributed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits unrestricted use, distribution, and reproduction in any medium, provided you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made.

Other papers from this open access journal are available free of charge from <http://www.springer.com/journal/41095>. To submit a manuscript, please go to <https://www.editorialmanager.com/cvmj>.